AMENDMENTS TO THE SPECIFICATION

Please amend the paragraph beginning at page 5, line 11 as follows:

Further referring to Figure 2, the quotient accumulator includes a first adder 42 that receives at one input an eighteen-bit quotient value QUOTIENT, and an eighteen bit input fed back from the accumulator output OUT. The first adder feeds a second adder 44 with the result of the summed QUOTIENT and OUT values. The second adder sums the QUOTIENT/OUT with a carry input from the remainder accumulator 50. The output from the second adder is shifted into a multi-bit register 46 clocked by input clock CLK. The output of the register OUT may then be used as the accumulator input. Because register 46 has a finite number of bits, the output will be computed with modular arithmetic. For example, if register 46 has N bits, the output will be computed modulo N.

Please amend the paragraph beginning at page 6, line 7 as follows:

The entire carry-generation process repeats every three (3) cycles (the DIVISOR value) to produce an accurate counter output OUT. As a result, the resolution of the clock is programmable to a very fine resolution, for example, to one hertz. Of course, the QUOTIENT, REMAINDER, and DIVISOR input values are entirely programmable by a user in establishing the desired frequency ratio. More generally, multiplexor 60 provides the accumulated value A as an input to register 54 in any clock cycle in which the value of A is less than the DIVISOR value B. However, in any cycle in which the accumulated value A exceeds the value B, multiplexor 60 provides the value of A minus B as an input to register 54. In this way, the value in register 54 increases either REMAINDER value using modular arithmetic with a modulus of B. The value of REM OUT thus may be described to increase modulo B.